

CLAIMS

What is claimed is:

1. A method for creating a plurality of arrays from a numbered list data set, the method comprising:
 - installing a program on a computer;
 - wherein the program performs steps comprising:
 - assigning an ID to a numbered list;
 - creating the arrays;
 - performing a Boolean OR operation on the ID and an array value; and
 - wherein the computer can determine whether an incoming numbered list is present in the numbered list data set by analyzing the arrays.
2. The method of claim 1 wherein the computer is a firewall.
3. The method of claim 2 wherein the program performs steps further comprising:
 - obtaining the numbered list data set comprising a plurality of the numbered lists; and
 - wherein the numbered lists comprise a plurality of numbers separated by decimals.
4. The method of claim 3 wherein the IDs are sequentially increasing powers of two.
5. The method of claim 4 wherein the quantity of arrays equals the quantity of numbers in the numbered lists.
6. The method of claim 5 wherein the arrays are MAXV fields long, MAXV being the maximum value of any number in the numbered lists or the incoming numbered list.
7. The method of claim 6 wherein the program performs steps further comprising:
 - responsive to the determination that the number is not the wildcard, translating the ID for the numbered list containing the number into a binary ID; and

responsive to the determination that the number is not the wildcard, translating the array value with an index equal to the number for the array associated with the number into a binary array value.

8. The method of claim 7 wherein the program performs steps further comprising:

determining whether one of the numbers is a wildcard;

responsive to the determination that the number is the wildcard, translating the ID for the numbered list containing the number into the binary ID; and

responsive to the determination that the number is the wildcard, translating a plurality of the array values for the array associated with the number into a plurality of the binary array values.

9. The method of claim 8 wherein the program performs steps further comprising:

translating the binary array value into a hexadecimal array value or a decimal array value.

10. A method for determining whether an incoming numbered list is present in a numbered list data set, the method comprising:

installing a program on a computer;

wherein the program performs steps comprising:

determining whether a number is the first number in the incoming numbered list;

responsive to the determination that the number is the first number in the incoming numbered list, setting a counter equal to an array value with an index equal to the number; and

wherein the computer can determine whether the incoming numbered list is present in the numbered list data set by analyzing the counter.

11. The method of claim 10 wherein the computer is a firewall.

12. The method of claim 11 wherein the program performs steps further comprising:

responsive to the determination that the number is not the first number in the incoming numbered list, performing steps comprising:

obtaining the array value with the index equal to the number;

performing a Boolean AND operation to generate a result; and

setting the counter equal to the result.

13. The method of claim 12 wherein the program performs steps further comprising:

determining whether the counter is equal to zero; and

responsive to the determination that the counter is equal to zero, indicating that the incoming numbered list is not present in the numbered list data set.

14. The method of claim 13 wherein the program performs steps further comprising:

responsive to the determination that the counter is not equal to zero, indicating that the incoming numbered list is present in the numbered list data set.

15. The method of claim 14 wherein the program performs steps further comprising:

obtaining the incoming numbered list; and

obtaining a plurality of numbered list data set arrays.

16. The method of claim 15 wherein the incoming numbered list comprises a plurality of the numbers separated by decimals.

17. The method of claim 16 wherein the program performs steps further comprising:

responsive to the determination that the number is not the first number in the incoming numbered list, performing steps comprising:

translating the counter into a binary counter;

translating the array value with the index equal to the number into a binary array value; and

translating the result into a hexadecimal result or a decimal result.

18. The method of claim 17 wherein the quantity of ones present in the binary counter indicates the location of the match between the incoming numbered list and the numbered list data set.

19. The method of claim 18 wherein the location of ones present in the binary version of the counter indicates the number of times the incoming numbered list is present in the numbered list data set.

20. The method of claim 19 wherein the program performs steps further comprising:

determining whether the number is a wildcard; and

responsive to the determination that the number is a wildcard, performing a Boolean OR operation between the counter and a value equal to sum of a plurality of IDs for the numbered lists in the numbered list data set.

21. A program product operable on a computer comprising:

a computer-usable medium;

wherein the computer usable medium comprises instructions for the computer to perform steps comprising:

assigning a plurality of IDs to a plurality of numbered lists;

creating a plurality of hashtables;

creating a wildcard array;

performing a Boolean OR operation on one of the IDs and a hashtable value;

and

wherein the computer can determine whether an incoming numbered list is present in a numbered list data set by analyzing the hashtables and the wildcard array.

22. The program product of claim 21 wherein the computer is a firewall.

23. The program product of claim 21 wherein the instructions further comprise:

obtaining the numbered list data set comprising a plurality of the numbered lists; and

wherein the numbered lists comprise a plurality of numbers separated by decimals.

24. The program product of claim 21 wherein the IDs are sequentially increasing powers of two.

25. The program product of claim 21 wherein the quantity of hashtables equals the quantity of numbers in the numbered lists.

26. The program product of claim 21 wherein the instructions further comprise:

responsive to the determination that a number is not a wildcard, translating the ID for the numbered list containing the number into a binary ID; and

responsive to the determination that the number is not the wildcard, translating a hashtable value with a key equal to the number for the hashtable associated with the number into a binary hashtable value.

27. The program product of claim 21 wherein the instructions further comprise:

determining whether a number is a wildcard;

responsive to the determination that the number is the wildcard, translating the ID for the numbered list containing the number into a binary ID; and

responsive to the determination that the number is the wildcard, translating a wildcard array value associated with the number position into a binary wildcard array value.

28. The program product of claim 21 wherein the instructions further comprise:

translating a binary hashtable value into a hexadecimal hashtable value or a decimal hashtable value.

29. The program product of claim 21 wherein the hashtable does not contain any zero values.

30. A program product operable on a computer comprising:

a computer-usable medium;

wherein the computer usable medium comprises instructions for the computer to perform steps comprising:

determining whether a number is the first number in an incoming numbered list;

responsive to the determination that the number is the first number in the incoming numbered list, setting a first counter equal to a hashtable value with a key equal to the number; and

wherein the computer can determine whether the incoming numbered list is present in the numbered list data set by analyzing the first counter.

31. The program product of claim 30 wherein the computer is a firewall.

32. The program product of claim 30 wherein the instructions further comprise:

responsive to the determination that the number is the first number in the incoming numbered list, performing steps comprising:

obtaining a wildcard array value with a key equal to the number position;

performing a Boolean OR operation on the first counter and the wildcard array value to generate a first result; and
setting the first counter equal to the first result.

33. The program product of claim 34 wherein the instructions further comprise:

responsive to the determination that the number is the first number in the incoming numbered list, performing steps further comprising:

translating the first counter into a binary first counter;
translating a wildcard array value with a key equal to the number into a binary wildcard array value; and
translating the first result into a hexadecimal first result or a decimal first result.

34. The program product of claim 30 wherein the instructions further comprise:

responsive to the determination that the number is not the first number in the incoming numbered list, performing steps comprising:

setting a second counter equal to a hashtable value with a key equal to the number;
obtaining a wildcard array value with a key equal to the number position;
performing a Boolean OR operation on the second counter and the wildcard array value to generate a first result;
performing a Boolean AND operation on the first result and the first counter to generate a second result; and
setting the first counter equal to the second result.

35. The program product of claim 36 wherein the instructions further comprise:

responsive to the determination that the number is not the first number in the incoming numbered list, performing steps further comprising:

translating the first counter into a binary first counter;

translating the second counter into a binary second counter;

translating the wildcard array value into a binary wildcard array value; and

translating the second result into a hexadecimal second result or a decimal

second result.

36. The program product of claim 30 wherein the instructions further comprise:

determining whether the first counter is equal to zero; and

responsive to the determination that the first counter is equal to zero, indicating that the incoming numbered list is not present in the numbered list data set.

37. The program product of claim 30 wherein the instructions further comprise:

responsive to the determination that the first counter is not equal to zero, indicating that the incoming numbered list is present in the numbered list data set.

38. The program product of claim 37 wherein the quantity of ones present in a binary first counter indicates the location of the match between the incoming numbered list and the numbered list data set.

39. The program product of claim 37 wherein the location of ones present in a binary first counter indicates the number of times the incoming numbered list is present in the numbered list data set.

40. The program product of claim 30 wherein the instructions further comprise:

obtaining the incoming numbered list; and

obtaining a plurality of numbered list data set hashtable.

41. The program product of claim 30 wherein the incoming numbered list comprises a plurality of numbers separated by decimals.
42. The program product of claim 30 wherein the instructions further comprise:
- determining whether the number is a wildcard; and
 - responsive to the determination that the number is a wildcard, performing a Boolean OR operation between the first counter and a value equal to sum of the IDs for the numbered lists in the numbered list data set.
43. An apparatus for determining whether an incoming numbered list is present in a numbered list data set, the apparatus comprising:
- means for obtaining the numbered list data set comprising a plurality of numbered lists;
 - wherein the numbered lists comprise a plurality of numbers separated by decimals;
 - means for assigning a plurality of IDs to the numbered lists;
 - wherein the IDs are sequentially increasing powers of two;
 - means for creating a plurality of arrays;
 - wherein the quantity of arrays equals the quantity of numbers in the numbered lists;
 - wherein the arrays are MAXV fields long, MAXV being the maximum value of any number in the numbered lists or an incoming numbered list;
 - means for determining whether one of the numbers is a wildcard;
 - responsive to the determination that the number is the wildcard, means for translating the ID for the numbered list containing the number into a binary ID;

responsive to the determination that the number is the wildcard, means for translating a plurality of array values for the array associated with the number into a plurality of binary array values;

responsive to the determination that the number is not the wildcard, means for translating the ID for the numbered list containing the number into the binary ID;

responsive to the determination that the number is not the wildcard, means for translating the array value with an index equal to the number for the array associated with the number into the binary array value;

means for performing a Boolean OR operation on the ID and the array value;

means for translating the binary array values into a hexadecimal array value or a decimal array value;

means for obtaining the incoming numbered list;

wherein the incoming numbered list comprises the plurality of numbers separated by decimals;

means for determining whether a number is the first number in the incoming numbered list;

responsive to the determination that the number is the first number in the incoming numbered list, means for setting a counter equal to the array value with the index equal to the number;

responsive to the determination that the number is not the first number in the incoming numbered list, the apparatus further comprises:

means for obtaining the array value with the index equal to the number;

means for translating the counter and the array value with the index equal to the number into binary numbers;

means for performing a Boolean AND operation to generate a result;

means for translating the result into a hexadecimal result or a decimal result;

means for setting the counter equal to the result;

means for determining whether the counter is equal to zero;

responsive to the determination that the counter is equal to zero, means for indicating that the numbered list is not present in the numbered list data set;

responsive to the determination that the counter is not equal to zero, means for indicating that the numbered list is present in the numbered list data set;

means for determining whether the number is the wildcard;

responsive to the determination that the number is the wildcard, means for performing the Boolean OR operation between the counter and a value equal to sum of the IDs for the numbered lists in the numbered list data set;

wherein the computer is a firewall;

wherein the quantity of ones present in the binary counter indicates the location of the match between the incoming numbered list and the numbered list data set; and

wherein the location of ones present in the binary counter indicates the number of times the incoming numbered list is present in the numbered list data set.